In the Claims

Claim 1, 8, 10, 11, 12, 13, 14, 15 and 16 are amended as follows.

1. (Currently amended) A method of managing contacts within a contact centre, comprising the steps of:

assigning to a received contact a priority and a skillset identifier, ~~whereby the received contact can be prioritised relative to other ones of said contacts~~;

creating a new software object for said received contact;

determining a queuing position for said new software object relative to at least one other software object representing a contact having a priority and a skillset identifier similar to said skillset identifier assigned to said received contact, <u>wherein said queuing position is determined from a comparison of the priority of the received contact and a priority of the contact represented by the at least one other software object</u>;

.. adding to said new software object a pointer to said at least one other software object; and

storing in memory a collection of said software objects each containing said pointer to at least one other of said software objects;

whereby said stored collection of said software objects provides a prioritised queue for a skillset, wherein the step of assigning to said received contact said priority and said skillset identifier includes assigning multiple said skillset identifiers, and wherein the step of adding said pointer comprises adding separate pointers to software objects in different queues.

2. (Previously presented) The method as claimed in claim 1, wherein said new software object includes pointers to two other software objects having said similar skillset identifier, said two other software objects representing the contacts immediately ahead of and behind said contact within a queue, except in the case in which the new software object is positioned at an end of said queue.

3. (Previously presented) The method as claimed in claim 1, further comprising the step of modifying said at least one other software object with a pointer to the new software object.

2

4. (Cancelled)

5. (Previously presented) The method as claimed in claim 1, further comprising the step of responding to a network request by sending over a network details of those software objects at a head of a queue matching criteria specified in the request.

6. (Previously presented) The method as claimed in claim 1, wherein the software objects are created and maintained by a contact manager, and a queuing module carries out said determination of said queuing position according to information associated with the new software object, the queuing module being further capable of adding said pointer to said new software object.

7. (Previously presented) The method as claimed in claim 6, wherein the contact manager has a memory space in which said software objects are stored, and the queuing module has a memory space in which said software objects are updated, and said memory spaces either form part of a common space.

8. (Currently amended) A method of distributing contacts across a network of contact centres, wherein each contact having an assigned priority is represented by a software object maintained at one of said contact centres, each said software object containing pointers to one or more other of said software objects maintained at the same contact centre to provide a prioritized queue of software objects at each said contact centre, wherein each software object is positioned in the prioritized queue in order of priority with respect to the one or more other software objects, and at least some of the software objects containing separate pointers to software objects in different queues wherein the method comprises:

upon a network resource having the capability of handling said contacts with certain criteria becoming available, requesting from each said contact centre the highest priority queued software object matching said criteria;

receiving information relating to each such highest priority queued software object from said contact centres;

determining which software object represents the contact with the highest priority and/or best match for the available resource; and

3

issuing routing instructions to cause said contact to be routed to the resource.

9. (Previously presented) The method as claimed in claim 8, wherein the contact centre which maintained the software object representing the selected contact carries out the further step of removing the selected software object from its queue and updating said software objects which contain said pointers to the selected software object, to thereby update the top of one or more of said queues represented at said contact centre by a collection of said software objects.

10. (Currently amended) A method of distributing contacts across a network of contact centres, wherein each contact having an assigned priority, is represented by a software object maintained at a contact centre, each said software object containing pointers to one or more said software objects maintained at the same contact centre to provide a prioritized queue of said software objects at each said contact centre, wherein each software object is positioned in the prioritized queue in order of priority with respect to the one or more other software objects, and at least some of the software objects containing separate pointers to software objects in different queues, wherein the method comprises:

maintaining a network queue of contacts by automatically replicating changes in contact software objects at each contact centre with corresponding changes in contact software objects in said network queue;

upon a network resource having the capability of handling contacts with certain criteria becoming available, determining from the network queue the highest priority queued software object matching said criteria; and

issuing routing instructions to cause said contact to be routed to the resource.

11. (Currently amended) A computer program product comprising a non-transitory computer-readable medium storing and/or recording instructions in machine readable form which when executed in a computer system for managing contacts at a contact centre are effective to cause the computer system to:

assign to a received contact a priority and a skillset identifier, whereby the received contact can be prioritised relative to other contacts;

create a new software object for said received contact;

4

determine a queuing position for said new software object relative to at least one other software object representing a contact having a similar skillset identifier, wherein said queuing position is determined from a comparison of the priority of the received contact and a priority of the contact represented by the at least one other software object;

add to said new software object a pointer to said at least one other software object, and

store in memory a collection of said software objects each containing said pointer to at least one other of said software objects;

whereby said stored collection of said software objects provides a prioritised queue for a skillset, wherein the assignment to said received contact of said priority and said skillset identifier includes assigning multiple said skillset identifiers, and wherein the adding of said pointer comprises adding separate pointers to software objects in different queues.

12.   (Currently amended) A computer program product arranged to distribute contacts across a network of contact centres, wherein each contact having an assigned priority is represented by a software object maintained at one of said contact centres, each said software object containing pointers to one or more other of said software objects maintained at the same contact centre to provide a prioritized queue of software objects at each said contact centre, wherein each software object is positioned in the prioritized queue in order of priority with respect to the one or more other software objects, and at least some of the software objects containing separate pointers to software objects in different queues, said computer program product comprising a non-transitory computer-readable medium storing and/or recording instructions in machine readable form which when executed in a computer system provided in a network of contact centres are effective to cause the computer system to :

upon a network resource having the capability of handling contacts with certain criteria becoming available, request from each said contact centre a highest priority queued software object representing a contact queued at said contact centre which matches said criteria;

receive information relating to each of said highest priority queued software objects from said contact centres;

determine which of said software objects represents the contact with the highest priority and/or best match for the available resource; and

issue routing instructions to cause said contact to be routed to the resource.

13. (Currently amended) A computer-implemented system for managing contacts in a contact centre, the system comprising:

a workflow processor for assigning to a received contact a priority and a skillset identifier, ~~whereby the received contact can be prioritised relative to other contacts~~;

an object creation module for creating a new software object for said contact;

a queuing manager for determining a queuing position for said new software object relative to at least one other software object representing a contact having a skillset identifier similar to said skillset identifier assigned to said received contact, wherein said queuing position is determined from a comparison of the priority of the received contact and a priority of the contact represented by the at least one other software object;

a software object modification module for adding to said software object a pointer to said at least one other software object, and

a memory for storing a collection of said software objects each containing a pointer to at least one other software object;

whereby said stored collection of said software objects provides a prioritised queue for a skillset, wherein the assignment to said received contact of said priority and said skillset identifier includes assigning multiple said skillset identifiers, and wherein the adding of said pointer comprises adding separate pointers to software objects in different queues.

14. (Currently amended) A computer-implemented system for distributing contacts across a network of contact centres, wherein each contact having an assigned priority is represented by a software object maintained at a contact centre, each said software object containing pointers to one or more other of said software objects maintained at the same contact centre to provide a prioritized queue of said software objects at each contact centre, wherein each software object is positioned in the prioritized queue in order of priority with respect to the one or more other software objects, and at least some of the software objects containing separate pointers to

software objects in different queues, wherein the system comprises <u>a computer, the computer comprising</u>:

a request generator for generating a request, upon a network resource having the capability of handling said contacts with certain criteria becoming available, said request being effective to determine from each contact centre the highest priority queued software object at said contact centre matching said criteria;

a network connection for forwarding said request to each contact centre and receiving therefrom information concerning the highest priority queued software object at each said contact centre matching said criteria;

comparison means for determining which of said software objects represents the contact with the highest priority and/or best match for the available resource; and

a routing instruction generator for issuing routing instructions to cause said contact to be routed to the resource.

15.   (Currently amended) <u>A computer program product comprising a non-transitory computer-readable medium storing and/or recording instructions in machine readable form which when executed in a computer system for managing contacts at a contact centre are effective to cause the computer system to create a software object</u> A software object embodied on a computer-readable medium, representing a contact <u>having an assigned priority</u> at a contact centre, said software object including: separate pointers to each of a plurality other of said software objects located immediately ahead of or behind said software object in each of a plurality of <u>prioritized</u> skillset queues, <u>wherein each software object is positioned in the prioritized skillset queues in order of priority with respect to the one or more other software objects,</u> the software object further comprising an identifier to the contact which the software object represents and a plurality of skillset identifiers enabling the software object to be identified in a search for software objects representing contacts which match given skillset criteria.

16.   (Currently amended) <u>A computer program product comprising a non-transitory computer-readable medium storing and/or recording instructions in machine readable form which when executed in a computer system for managing contacts at a contact centre are effective to cause the computer system to create a</u>[[A]] virtual queue of contacts embodied on a computer-readable medium wherein: each contact <u>has an</u>

7

assigned priority ~~within the queue~~ and is represented within the queue by a software object including separate pointers to each of a plurality of other said software objects located immediately ahead of or behind said software object in each of a plurality of prioritized skillset queues, wherein each software object is positioned in the prioritized skillset queues in order of priority with respect to the one or more other software objects, the software object further comprising an identifier to the contact which the software object represents and a plurality of skillset identifiers enabling the software object to be identified in a search for said software objects representing said contacts which match given skillset criteria, and wherein the order of said contacts within the queue is determinable from the aggregated pointers between said software objects.

17.   (Previously presented) The method as claimed in claim 6, wherein the contact manager has a contact manager memory space, and the queuing module has a queuing module memory space, and wherein each of said software objects is stored in two corresponding copies, a first of said copies being stored in the queuing module memory space and a second of said copies being stored in the contact manager memory space, and wherein a replication service is provided which is configured to ensure that changes to the first of said copies are reflected in corresponding changes to the second of said copies, and that changes to the second of said copies are reflected in corresponding changes to the first of said copies.